DONALD D ANTONELLI
DAVID T TERRY
MELVIN BARNES
WILLIAM I SOLOMON*
GREGORY E MONTONE
RONALD J SHORE
DONALD E STOUT
ALAN E SCHIAVELLI
JAMES N DRESSER
CARL I BRUNDIDGE*

RANDALL S SVIHLA
DAVID S LEE*
PAUL J SKWIERAWSKI*
ROBERT M BAUER*
SAMUEL W NTIROS
DEMETRA J MILLS

*ADMITTED OTHER THAN VA

LAW OFFICES
ANTONELLI, TERRY, STOUT & KRAUS, LLP
SUITE 1800
1300 NORTH SEVENTEENTH STREET
ARLINGTON, VIRGINIA 22209

PATENT AGENT
LARRY N ANAGNOS

TELEPHONE
(703) 312-6600
FACSIMILE
(703) 312-6666
E-MAIL
email@antonelli.com

March 19, 1998

Honorable Commissioner of
Patents and Trademarks
Washington, D.C. 20231

Attorney Docket Number: 500.36133X00

Sir:

Attached please find the application papers of Takayoshi SHIMOKAWA, Kazuo MASAI, covering new and useful improvements in DATA STRUCTURE IN DATABASE, DATABASE SYSTEM FOR MANAGING DATABASE AND DATABASE MANAGING METHOD AND SYSTEM, comprising:

Specification, Twenty (20) Claims and Abstract of the Disclosure (31 pages)

English language, Combined Declaration and Power of Attorney (2 pages)

Eleven (11) Sheets of Drawings Showing Figures 1-13

Letter Claiming Right of Priority and Certified Copy of Japanese Patent Application No. 9-065919

Assignment and Recording of Assignment Letter

U.S. Government Filing Fee of $1,036.00

U.S. Government Recording Fee of $40.00

Change of Correspondence Address

Information Disclosure Sheet Under 37 CFR 1.56 With Attached References

Please charge any shortages in the fees or credit any overpayments thereof the deposit account of Antonelli, Terry, Stout & Kraus, LLP Account No. 01-2135 (500.36133X00).

Respectfully submitted,

ANTONELLI, TERRY, STOUT & KRAUS, LLP

Carl I. Brundidge
Registration No. 29,621

CIB/nac
Attachments

Please type a plus sign (+) inside this box → [+]

PTO/SB/122 (11-96)
Approved for use through 6/30/99. OMB 0651-0035
Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

CHANGE OF CORRESPONDENCE ADDRESS

Application

Address to:
Assistant Commissioner for Patents
Washington, D.C. 20231

Application Number	
Filing Date	March 19, 1998
First Named Inventor	T. SHIMOKAWA
Group Art Unit	
Examiner Name	
Attorney Docket Number	500.36133X00

Please change the Correspondence Address for the above-identified application to:

[XX] Customer Number [020457] →

Type Customer Number here

PATENT & TRADEMARK OFFICE

020457

OR

[ ] Firm *or* Individual Name					
Address					
Address					
City		State		ZIP	
Country					
Telephone		Fax			

This form cannot be used to change the data associated with a Customer Number. To change the data associated with an existing Customer Number use "Request for Customer Number Data Change" (PTO/SB/124).

I am the :

[ ] Applicant.

[ ] Assignee of record of the entire interest.
Certificate under 37 CFR 3.73(b) is enclosed.

[XX] Attorney or agent of record .

Typed or Printed Name	Carl I. Brundidge	Registration NO. 29,621
Signature		
Date	March 19, 1998	

Burden Hour Statement: This form is estimated to take 0 ' hours to complete. Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to comple e this form should be sent to the Chief Information Officer, Patent and Trademark Office, Washington, DC 20231 DO NOT SEND FEES OR COM LETED FORMS TO THIS ADDRESS. SEND TO. Assistant Commissioner for Patents, Washington, DC 20231

DATA STRUCTURE IN DATABASE, DATABASE SYSTEM FOR MANAGING DATABASE AND DATABASE MANAGING METHOD AND SYSTEM

BACKGROUND OF THE INVENTION

The present invention relates to a time series database processing system of especially ultra-large scale for storing data pieces serving as updating detailed information in sequence of time series in a database and for controlling addition/deletion/retrieval of data.

When data pieces are loaded on a database of large scale and a specified data piece is retrieved from the database, index is generally applied. Indexing is effective when an item serving as a key during retrieval can be specified. The indexing is a contrivance in which specified key items of a database are collected, a pointer is provided over the key items to take the form of a balanced tree (B tree), and the tree can be traced at a high speed up to a location corresponding to a leaf of the tree in accordance with information indicating which range a key of a specified value lies in. "An Introduction to Database Systems, 3.4 Indexing" by C. J. Date, Addison-Wesley, 1986, pp.68-77 teaches a contrivance in which information corresponding to storage locations of all data items can be obtained for all the data items. If the database is for about million cases or events, there occurs no problem but in a database of ultra-large scale for

billion cases or trillion cases, however, the maintenance of index per se swells and especially, keys which are added in time series fashion may not be handled well.

When data pieces are added in time series fashion, the indexing grows in a direction in which time increases, as shown in Fig. 1. Further, from the standpoint of deletion, it is known that as deletion of indices for which a constant time has expired proceeds, data pieces remain at only one side portion of the indexing tree and values of items are lost in spite of the existence of nodes on the other side portion, thereby placing the indexing in very inefficient condition. In such an event, it is necessary that the indexing be reconstructed by a technique called reorganization to delete wasteful areas in the indexing and promote the efficiency. But in the time series database of ultra-large scale, this is not practical because work far exceeding the permissible range is required.

A utility for data loading uses a technique for writing data directly to a physical area of a database and therefore, with this utility, data can be written at a high speed. However, the utility for high-speed data loading generally inhibits direct data write to the physical area during data loading from conflicting area at other retrieval or updating access. In other words, data load shall compulsorily be executed while inhibiting access to a specified table for retrieval/updating or a part of a table

for retrieval/updating. This forces retrieval of the database to be once stopped each time that time series data added almost every day is loaded. In a database of ultra-large scale, it takes one day or more for retrieval per se in some applications. In that case, data loading cannot be permitted unless retrieval is stopped, leading to fatal inconvenience. To avoid such situations, data can be added through usual data insertion operation without resort to data loading but in this case, the performance is degraded by approximately by one order as compared to data loading of physical write type and besides, locking must be acquired for concealing data during addition, largely affecting the performance of operation for retrieval of all cases or events in the database.

In order to delete a data piece in the database for which a constant time has expired, the data piece is typically required to be retrieved and even in the presence of index, time comparable to that for inserting data piece by piece is consumed. In the absence of index, all data pieces are retrieved for the purpose of deleting a data piece of interest and consequently, in the database of ultra-large scale, it takes one day or more to operate only the deletion processing and practically, the time series database cannot be materialized.

Thus, for the deletion of data for which a constant time has expired, time exceeding that for retrieval of all pieces of data is consumed in the absence of index but conversely, in the presence of index, indexing

is updated during deletion, leading to an operation which consumes much time as in the case of data insertion. Accordingly, it is practically difficult to realize daily data deletion for the database which takes one day or more to retrieve all data pieces.

SUMMARY OF THE INVENTION

An object of the invention is to provide method and system which can eliminate conflict of the operation of time series data loading and data deletion with the operation of data retrieval in a database system and which can mitigate suppression imposed on retrieval by the system.

Another object of the invention is to provide a database managing system which can dispense with reorganization of an index tree which loses balance owing to addition of time series data.

According to the present invention, there is provided a database managing method for managing data pieces in a database, comprising the steps of:

adding, to a given time series data piece for a predetermined time, book mark information having bookmark information indicative of the corresponding time and state transition information indicative of a state of the time series data piece for the predetermined time;

providing, as the state transition information, one of a value indicative of an online state in which a data area is permitted to be retrieved, a value indicative

of a loading state in which loading of data in the data area has not yet been completed and the data area is not permitted to be retrieved, and a value indicative of an empty state in which data in the data area is empty; and

loading time series data pieces for the predetermined time in a plurality of data areas in the database in sequence of times corresponding to the time series data pieces.

The method further comprises the steps of:

reading, from the plurality of data areas, a plurality of bookmark information pieces each having state transition information and bookmark information in accordance with a data retrieval request applied to the database by designating a time; and

detecting the bookmark information including the designated time and when the state transition information included in the detected bookmark information indicates the online state, reading a time series data piece corresponding to the detected bookmark information.

When the state transition information included in the detected bookmark information indicates either a value indicative of the loading state or a value indicative of the empty state, it can be decided that the data retrieval request is not responded to.

The method further comprises the steps of:

reading, from the plurality of data areas, a plurality of bookmark information pieces each having state transition information and bookmark information in

accordance with a data deletion request applied to the database by designating a time; and

detecting the bookmark information including the designated time and when the state transition information included in the detected bookmark information indicates the online state, setting a value indicative of the empty state in the state transition information included in the detected bookmark information.

The method further comprises the steps of:

cumulating repeatedly applied time series data pieces in a cumulative data storage area until they reach total data for the predetermined time; and

after the repeatedly applied time series data pieces have been collected up to the total data for the predetermined time, adding, to a data piece in the cumulative data storage area, bookmark information having bookmark information indicative of a time corresponding to the data piece for the predetermined time and state transition information indicative of a state of the data piece for the predetermined time and loading resulting data pieces in the plurality of data areas in the database in sequence of times corresponding to the time series data pieces.

According to the present invention, a data structure realized in a database comprises:

a plurality of data areas for loading given time series data pieces at predetermined locations of the database in sequence of times; and

a predetermined bookmark information area having bookmark information indicative of a time corresponding to a time series data piece loaded in each of the data areas and state transition information indicative of a state of the data piece in each data area,

wherein the state transition information has one of a value indicative of an online state in which the data area is permitted to be retrieved and a value indicative of a loading state in which loading of data in the data area has not yet been completed and the data area is not permitted to be retrieved. The data pieces are arranged consecutively in the database while having a predetermined data capacity so that the plurality of bookmark information areas in the plurality of data areas may be read consecutively.

In the present invention, the database is divided into segments which are each minimum blocks for storage area management and time series data pieces are stored in the segments. When data is loaded on the database, a time at which the data is loaded is stored as a bookmark at a predetermined location in a start segment from which the addition starts with the database. Thanks to the bookmark, when retrieval of time designation or time interval designation is carried out, the retrieval range can be narrowed physically by utilizing the bookmark.

When data loading is effected, the database can be brought into a loading unfinished state by locating the

bookmark in other place than the place in which the data is being loaded. Consequently, data can be loaded directly on a physical segment without affecting other retrieval. At the time that the data loading is completed, the bookmark is written in the above other place and the database is recognized by such assigning a bookmark thereto.

In the case of data deletion, when data pieces following a specified bookmark are to be deleted collectively, the areas are effectively emptied changing the bookmark for the unit of segment within a short time without actually accessing to the data. By managing the areas of the database in a unit of segment in wrap-around fashion, the always pooled consecutive areas can be used from one side to load data and replenish an area from the other side of the consecutive areas.

The present invention is effective for a computer system having a database and especially for a database system for retrieval in which data pieces reach the database system in sequence of time series and data change other than addition or insertion and deletion of time series data is not carried out.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 is a diagram showing indices of a tree which loses balance owing to addition/deletion of time series data.

Fig. 2 is a diagram showing an embodiment of system construction according to the present invention.

Fig. 3 is a diagram showing the construction of an embodiment of a storage apparatus of the present invention.

Fig. 4 is a diagram for explaining a bookmark information area.

Fig. 5 is a flow chart of an embodiment of the retrieval processing.

Fig. 6 is a diagram showing the state of the storage apparatus to explain the flow chart of Fig. 4.

Fig. 7 is a flow chart showing an embodiment of the data load processing.

Fig. 8 is a diagram showing the state of the storage apparatus in mid course to explain the flow chart of Fig. 6.

Fig. 9 is a diagram showing the final state of the storage apparatus to explain the flow chart of Fig. 6.

Fig. 10 is a flow chart of an embodiment of the deletion processing.

Fig. 11 is a diagram showing the state of the storage apparatus in mid course to explain the flow chart of Fig. 9.

Fig. 12 is a diagram showing the construction of the storage apparatus to explain an embodiment of a wrap-around architecture.

Fig. 13 is a diagram showing the construction of another embodiment of the storage apparatus of the present invention.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

The present invention will now be described by way of example with reference to the accompanying drawings.

Referring to Fig. 2, there is illustrated an embodiment of system construction according to the present invention.

As shown in Fig. 2, a database system principally comprises a database system apparatus 10 having a central processing unit (CPU) 11 and a storage apparatus 13 for physically storing data. A database management program 12 operates on the system apparatus 10 to store actual data from a cumulative data area 8 onto the storage apparatus 13. Provided in the storage apparatus 13 are a data area 14 and a system definition information area 15 for storing definition information of data. The area 8 may have a data entity area 8A and an empty area 8B in order to store time series data pieces for a predetermined time and transfer the stored data to the storage apparatus 13.

Referring to Fig. 3, the construction of the storage apparatus is shown in greater detail to give a detailed explanation of the system definition information area 15 and data area 14. In the present embodiment, the data area 14 has consecutive areas secured on the storage apparatus 13 so as to be divided into management blocks called segments 20. Data pieces generated in time series fashion are put together in the area 8 by means of the management program 12 until they reach an amount for a

constant time. The collected data pieces are stored in one of the management block segments of the consecutive areas of the database in the form of the storage apparatus 13, along with a time for storage which is read out of a clock 9 and stored in the same segment or otherwise at a different location. The segment 20 includes, for example, a data storage area 21 for storing real data and a bookmark information area 22 for storing management information for the data stored in the data storage area 21. In the present embodiment, the segment 20 consists of a plurality of pages each being a unit of disk input/output.

The system definition information 15 has information for managing the storage location of time series data, including information for pointing a segment 20 which is the oldest in time series and information for pointing the start of an empty segment area.

As shown in Fig. 4 useful to explain the bookmark information area 22, the bookmark information area includes a time information area 23 for storing information concerning a time which is specific to data stored in the segment 20 and which is delivered out of the clock 9 and a status flag area 24 for storing status flag information indicative of a shifting or transition state (to be described below) of the segment 20. The shifting or transition state is classified into three states or modes including "online" indicating that the data storage area is accessible, "loading" indicating that data is now being

inserted and "empty" indicating that no data is present in the data storage area. The status of the segment 20 shifts from one mode or state to another.

Next, the operation of the present embodiment will be described.

In the time series database, retrieval for which time is specified is frequently practiced. For example, the title and the date of issue of a book published by a publisher are stored in time series fashion in a time series database of the publisher by using the issue date as a key and an instance will be described hereunder in which the database is retrieved for a list of titles of books issued over three months which range from March, 1994 to May, 1994.

The retrieval processing of the present embodiment will be described with reference to Figs. 5 and 6. Fig. 5 is a flow chart showing an embodiment of the retrieval processing in the present embodiment and Fig. 6 is a diagram showing the state of the storage apparatus useful to explain the flow chart of Fig. 5.

In the database system of the present embodiment, information for pointing a segment 20 which stores the oldest data in time series is first acquired from the system definition information 15 (step 500). Then, the database system acquires time information t (February, 1994) and status information (online) from a bookmark information area 22 of the pointed segment 20 (step 501). Acquisition of the system definition information is carried

out at a high speed because a predetermined capacity of data can be acquired starting with the start of a plurality of segments arrayed at equi-capacity intervals on the database.

If the acquired status information is "empty" or "loading", the data to be retrieved has not been stored in the segment 20 or data is now being inserted in the segment 20 and hence it is determined that access is impossible and the retrieval processing ends (step 502).

If the status information is "online", access is permitted and the program proceeds to the next process (step 503). The posterior retrieval request time (May, 1994) is compared with the time information (February, 1994) stored in the bookmark information area 22 to decide whether the intended data is stored in the database. If the result of comparison is "Yes", in a test whether the stored newer data is newer than the range of the retrieval object (March, 1994 to May, 1994), the retrieval processing ends. When "No" is issued in the decision process, the program proceeds to the next process (step 504) to decide whether the segment 20 now pointed is within the retrieval request time (March, 1994 to May, 1994). Since the segment 20 is of February, 1994, this data storage area 21 is excluded from the retrieval object and a segment 20 for storing data which succeeds in terms of time series is pointed (step 506). For example, it is assumed that a magnetic disk device is used as the storage apparatus 13

and given that all of the segments 20 have the same size, the succeeding segment can be pointed by moving the size of segment (a moving amount relative to the magnetic head) starting from the header of the present disk.

Next, for that succeeding segment 20, the decision process like the above (steps 502, 503 and 504) is executed. When it is determined in the process (step 504) that the segment 20 is one which meets the retrieval request, data is read out of the corresponding data storage area 21 in the segment 20 (step 505). Since the header of the disk points the start of a segment 20 which stores the next data in terms of time series after the data has been read out of the data storage area 21 (step 506), time information is again acquired from a bookmark information area 22 and thereafter, the decision is repeated in a similar way. In this manner, the segments 20 are sequentially read. Since in the decision process (step 503) of a segment 20 the segment is determined to be outside the retrieval object, the retrieval processing ends at that time.

Next, the data load processing will be described with reference to Fig. 7. Fig. 7 is a flow chart showing the data load processing in the present embodiment. In the present embodiment, an instance will be described in which data pieces of from July, 1994 to August, 1994 are loaded from the system apparatus to the database, that is, data loading is carried out. It is now assumed that data pieces to be inputted in the form of files have already been

sorted in terms of time series. The following description will be given by referring to an example where data is added to the initial state illustrated in Fig. 6.

Firstly, empty segment information is read out of the system definition information 15 (step 600). An empty segment 20 is pointed by that information. In order to read input data, the input file is accessed and data (July, 1994) is read (step 602). Because of the presence of the data, "presence" is determined in the process (step 602) and the program proceeds to the process (step 603). In the process (step 603), a write process is executed. Firstly, the time, information (July, 1994) is written at the time information area and a flag "loading" indicating currently loading at the status flag area in the bookmark information area 22, and data is written into the data storage area 21. After completion of write, a state as shown in Fig. 8 prevails.

After write of data for one segment has been terminated, the database system reads the next input data from the file (step 601). Because of the presence of data for August, 1994, "presence" is determined in the decision process (step 602). Through the same logic as that used for writing the data for July, 1994, time information (August, 1994), a status flag "loading" and data are written at the time information area 23, status flag area 24 and data storage area 21 in a segment 20 (step 603).

After completion of write, the system is about to read the next data from the file (step 601). But, since

data has already been absent in the file, "absence" is determined in the decision process (step 602) and the program proceeds to the next process (step 604 in Fig. 7).

After write of the input data to the database has been finished, the database system starts updating the status flag in the bookmark information area in order to make the segments written with the new data accessible (step 604).

When write of the final data is completed, the database system reads the empty segment information 16 in the system definition information 15 and points a segment 20 which has initially been written with the new data. Since in that segment 20 the status flag in the bookmark area 22 is set with "loading", this flag is shifted to "online". This permits that segment to be retrieved. In the present embodiment, the size of segment is defined as in the case of retrieval and therefore, a segment 20 stored with the next information in time series fashion can be pointed.

The shift or transition processing from "loading" to "online" ends when the status flag of the read bookmark information area indicates "empty" and address information for that segment is set in the empty segment information 16 in the system definition information 15 (step 605). A state in which the data load processing is thoroughly completed is shown in Fig. 9. As will be seen from the above, even during loading, the database system need not

suppress the data retrieval request because by adopting the flag, it is possible to realize such a setting operation that access to the disk having a segment in which the "loading" flag is not raised can be permitted and access to the disk having a segment in which the flag is raised cannot be permitted.

Next, the deletion processing will be described with reference to Fig. 10. Fig. 10 is a flow chart showing an embodiment of the deletion processing.

In the present embodiment, the state shown in Fig. 6 is considered as the initial state and the segment 20 for February, 1994 is deleted.

Firstly, start segment information 16 is read out of the system definition information 15 (step 700). Time information (February, 1994) is acquired from the bookmark information area 22 of the segment 20 and it is decided whether the segment 20 is one which is an object to be deleted (step 701).

Since the deletion object is of February, 1994, that segment 20 is determined to be the deletion object. The start segment information 16 in the system definition information 15 is shifted to the next segment 20 (for March, 1994) in time series fashion. The segment size is determined and therefore, a start segment address can be obtained by adding by the segment size (step 702).

Subsequently, time information (null) is set to the bookmark information area 22 (step 703) and "empty" is set to the status flag (step 704). By initializing the

bookmark information area 22 (steps 703 and 704), the segment 20 can be shifted to an inaccessible state.

A segment 20 which is next in terms of time series is pointed (step 705) and time information (March, 1994) is acquired from the bookmark information area 22 of that segment 20. The acquired time information (March, 1994) is compared with February, 1994 for the deletion object and it is determined that the segment 20 is not the deletion object (step 701), thus ending the deletion processing. After the completion, the database assumes a state as shown in Fig. 11.

In the present deletion processing, internal data need not be directly accessed and only the bookmark information area is taken as the object, thereby making it possible to perform deletion within a short time and during online.

The segments are used in wrap-around fashion to attain an advantage that no reorganization is needed even when addition/deletion is repeated. Finally, the wrap-around architecture will be described.

Referring now to Fig. 12, there is illustrated an embodiment of the wrap-around architecture. A method of wrap-around which uses the respective segments temporally cyclically can be realized by setting a "start" flag 26 and a start address area 25 in the bookmark information area 22 of each segment 20. In a segment which is at the physically lowest position, "1" is set in the "start" flag

26 and an address of a start one 20 of the segments is set in the start address area 25. Even in the processing of retrieval/deletion/insertion, this setting can be realized easily by adding a process of jumping to the start address on the extension of the retrieval/deletion/insertion processing because the processing of referring to the bookmark information area is always employed in the retrieval/deletion/insertion processing. In this example, a database is shown which always holds data of the latest six months in a minimal segment capacity.

Data pieces over a certain constant time are frequently managed by a plurality of segments 20. Fig. 13 shows an embodiment of the present invention which meets this case. In the present embodiment, a system is available in which bookmark information pieces are stored in a bookmark information area 22' in the system definition information 15 so as to undergo centralized control. This system is more practical because it has such a merit that the area to be written with data is not limited by the bookmark information area and the respective segments need not have capacities which are matched to the same value.

As described above, according to the embodiments of the present invention, the intended data can be accessed without resort to index by retrieving thoroughly only the specified control information storage range without retrieving the whole of the database.

In an embodiment of the present invention, data loading can be accomplished at a very high speed without

stopping retrieval by temporarily making addition of data to a different empty segment in advance and at the time of completion of the data loading, assigning the data with a bookmark in the form of a table of the database.

In an embodiment of the present invention, in connection with deletion of data for which a constant time is exceeded, a segment to be deleted can be specified by retrieving the bookmark and the segment is a unit of area management of the database so that the area may be emptied, with the result that deletion can be accomplished within a very short time (typically, approximately several seconds to several minutes).

According to the present invention, the scale of the bookmark information can be small as compared to the data amount which is very large, thus ensuring that the maintenance processing can be realized very easily and the bookmark information can be retrieved within a very short time even in a large-scale database.

According to the present invention, in a large-scale database which has a very large amount of data and in which storage and deletion of data pieces which arrive in sequence of time series, high-speed retrieval can be carried out and even during online, the data load and deletion processing can be realized.

CLAIMS:

1. A data structure realized in a database comprising:

a plurality of data areas in which given time series data pieces are loaded in sequence of times; and

bookmark information areas respectively provided at predetermined locations in said plurality of data areas and each having a pair of bookmark information indicative of a time corresponding to a time series data piece loaded in each of said data areas and state transition information indicative of a state of the data piece in each data area, said state transition information being allowed to have one of a value indicative of an online state in which the data area is permitted to be retrieved and a value indicative of a loading state in which loading of data in the data area has not yet been completed and the data area is not permitted to be retrieved.

2. A data structure according to claim 1, wherein said plurality of data areas have each a predetermined data capacity and are arranged consecutively in said database in order that said plurality of bookmark information areas in said plurality of data areas can be read consecutively.

3. A data structure according to claim 1, wherein the state transition information in at least one of said plurality of data areas is allowed to have one of a value indicative of an online state in which the data area is permitted to be retrieved, a value indicative of a loading

state in which loading of data in the data area has not yet been completed and the data area is not permitted to be retrieved and a value indicative of a state in which data in the data area is empty.

4. A data structure realized in a database comprising:

a plurality of data areas in which given time series data pieces are loaded at predetermined locations, respectively, in said database in sequence of times; and

predetermined bookmark information areas each having a pair of bookmark information indicative of a time corresponding to a time series data piece loaded in each of said data areas and state transition information indicative of a state of the data piece in each data area, said state transition information having one of a value indicative of an online state in which the data area is permitted to be retrieved and a value indicative of a loading state in which loading of data in each data area has not yet been completed and the data area is not permitted to be retrieved.

5. A data structure according to claim 5, said plurality of data areas have each a predetermined data capacity and are arranged consecutively in said database in order that said plurality of bookmark information areas in said plurality of data areas can be read consecutively.

6. A data structure according to claim 4, wherein the state transition information in at least one of said plurality of data areas has a value indicative of an online

state in which the data area is permitted to be retrieved, a value indicative of a loading state in which loading of data in the data area has not yet been completed and the data area is not permitted to be retrieved and a value indicative of a state in which data in at least one data area is empty.

7. A database managing method for managing data in a database comprising the steps of:

adding, to a predetermined location in a given time series data piece for a predetermined time, bookmark information having bookmark information indicative of a time corresponding to said time series data piece for said predetermined time and state transition information indicative of a state of said time series data piece for said predetermined time;

providing, as said state transition information, one of a value indicative of an online state in which the data area is permitted to be retrieved, a value indicative of a loading state in which loading of data in the data area has not yet been completed and the data area is not permitted to be retrieved and a value indicative of a state in which data in the data area is empty; and

loading time series data pieces for predetermined times in a plurality of data areas in said database in sequence of times corresponding to said time series data pieces.

8. A database managing method according to claim 7 further comprising the steps of:

reading, from said plurality of data areas, a plurality of bookmark information pieces each having state transition information and bookmark information in accordance with a data retrieval request applied to said database by designating a time; and

detecting the bookmark information including said designated time and when the state transition information included in said detected bookmark information indicates said online state, reading a time series data piece corresponding to said detected bookmark information.

9. A database managing method according to claim 7 further comprising the step of:

when the state transition information included in said detected bookmark information indicates either a value indicative of said loading state or a value indicative said empty state, determining that said data retrieval request is not responded to.

10. A database managing method according to claim 7 further comprising the steps of:

reading, from said plurality of data areas, a plurality of bookmark information pieces each having state transition information and bookmark information in accordance with a data deletion request applied to said database by designating a time; and

detecting the bookmark information including said designated time and when the state transition information included in said detected bookmark information indicates said online state, setting a value indicative of an empty

state in said state transition information included in said detected bookmark information.

11. A database managing method according to claim 7 further comprising the steps of:

cumulating repeatedly applied time series data pieces in a cumulative data storage area until they reach total data for said predetermined time; and

after said repeatedly applied time series data pieces have been collected up to said total data for said predetermined time, adding, to a data piece in said cumulative data storage area, bookmark information having bookmark information indicative of a time corresponding to the data piece for said predetermined time and state transition information indicative of a state of said time series data piece for said predetermined time and loading resulting data pieces in said plurality of data areas of said database in sequence of times corresponding to said time series data pieces.

12. A database managing method for managing data in a database comprising the steps of:

adding, to a predetermined location in a given time series data piece for a predetermined time, bookmark information having bookmark information indicative of a time corresponding to said time series data piece for said predetermined time and state transition information indicative of a state of said time series data piece for said predetermined time and start area information having a

flag indicating whether the area is the final one of a plurality of areas in said database and an address area for setting an address;

providing, as said state transition information, one of a value indicative of an online state in which the data area is permitted to be retrieved and a value indicative of a loading state in which loading of data in the data area has not yet been completed and the data area is not permitted to be retrieved;

loading time series data pieces for predetermined times in a plurality of consecutive data areas in said database in sequence of times corresponding to said time series data pieces; and

raising said flag of start area information in the final one of said plurality of consecutive data areas and setting an address of first one of said plurality of consecutive data areas in said address area.

13. A database managing method according to claim 12 further comprising the steps of:

adding, to a predetermined location in a time series data piece for a predetermined time applied so as to be loaded in said database, bookmark information having bookmark information indicative of a time corresponding to said time series data piece for said predetermined time and state transition information indicative of a state of said time series data piece for said predetermined time;

reading all state transition information pieces in said database to detect bookmark information having the

oldest time and loading said time series data piece for said predetermined time applied so as to be loaded in said database in a data area corresponding to said oldest bookmark information; and

updating said oldest bookmark information to said bookmark information corresponding to said loaded data.

14. A database managing method for managing data in a database comprising the steps of:

reading bookmark information having bookmark information indicative of a time corresponding to a given time series data piece for a predetermined time and state transition information indicative of a state of said time series data piece for said predetermined time from a predetermined bookmark area and setting the state of said time series data piece in said state transition information to a value indicative of a state in which data is empty so as to write said bookmark information in said database;

loading given time series data pieces for given predetermined times in a plurality of data areas in said database in sequence of times corresponding to said time series data pieces; and

after the step of loading said data pieces has been completed, writing bookmark information having bookmark information indicative of a time corresponding to a time series data piece for said predetermined time and state transition information indicative of an online state of said time series data piece for said predetermined time in said predetermined bookmark area.

15. A database managing method according to claim 14 further comprising the step of loading time series data pieces for predetermined times in a plurality of data areas in said database in sequence of times corresponding to said time series data pieces.

16. A database managing method according to claim 15 further comprising the steps of:

reading, from said plurality of data areas, a plurality of bookmark information pieces each having state transition information and bookmark information in accordance with a data retrieval request applied to said database by designating a time; and

detecting the bookmark information including said designated time and when the state transition information included in said detected bookmark information indicates said online state, reading a time series data piece corresponding to said detected bookmark information.

17. A database managing method according to claim 14 further comprising the step of:

when the state transition information included in said detected bookmark information indicates either a value indicative of said loading state or a value indicative of said empty state, determining that said data retrieval request is not responded to.

18. A database managing method according to claim 14 further comprising the steps of:

reading, from said plurality of data areas, a plurality of bookmark information pieces each having state

transition information and bookmark information in accordance with a data deletion request applied to said database by designating a time; and

detecting the bookmark information including said designated time and when the state transition information included in said detected bookmark information indicates said online state, setting a value indicative of said empty state in said state transition information included in said detected bookmark information.

19. A database managing method according to claim 14 further comprising the steps of:

cumulating repeatedly applied time series data pieces in a cumulative storage area until they reach total data for said predetermined time; and

after said repeatedly applied time series data pieces have been collected up to said total data for said predetermined time, adding, to a data piece in said cumulative data storage area, bookmark information having bookmark information indicative of a time corresponding to said data piece for said predetermined time and state transition information indicative of a state of said time series data piece for said predetermined time and loading resulting data pieces in said plurality of data areas in said database in sequence of times corresponding to said time series data pieces.

20. A database managing system comprising:

a processor having a memory for storing given time series data pieces for predetermined times and a clock

for reading times at which said time series data pieces are applied; and

a database connected to said processor and having bookmark information indicative of a time corresponding to a time series data piece for a predetermined time, state transition information indicative of a state of said time series data piece of said predetermined time and said time series data pieces for said predetermined times, said state transition information having one of a value indicative of an online state in which the data area is permitted to be retrieved, a value indicative of a loading state in which loading of data in the data area has not yet been completed and the data area is not permitted to be retrieved and a value indicative of a state in which data in the data area is empty.

ABSTRACT OF THE DISCLOSURE

Bookmark information having bookmark information indicative of a time corresponding to a given time series data piece for a predetermined time and state transition information indicative of a state of the time series data piece for the predetermined time is loaded, along with the time series data piece, in a database. The state transition information has one of a value indicative of an online state in which the data area is permitted to be retrieved, a value indicative of a loading state in which loading of data in the data area has not yet been completed and the data area is not permitted to be retrieved and a value indicative of a state in which data in the data area is empty. The time series data pieces for the predetermined times are loaded in a plurality of data areas of the database in sequence of times. Even during deletion or addition of data, all data retrieval requests need not be suppressed. In accordance with a data deletion request, state transition information corresponding to a data piece of interest is set to a value indicating that the data piece of interest is empty. For data retrieval, the state transition information can be read from a storage area being in online condition. Degradation of the performance of a database system caused by B tree indexing eccentrically extending in one direction owing to addition of time series data can be prevented.

FIG. 1

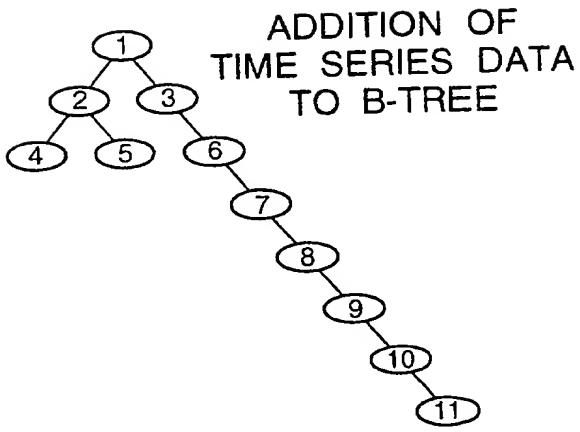

ADDITION OF
TIME SERIES DATA
TO B-TREE
1
2
3
4
5
6
7
8
9
10
11

FIG. 2

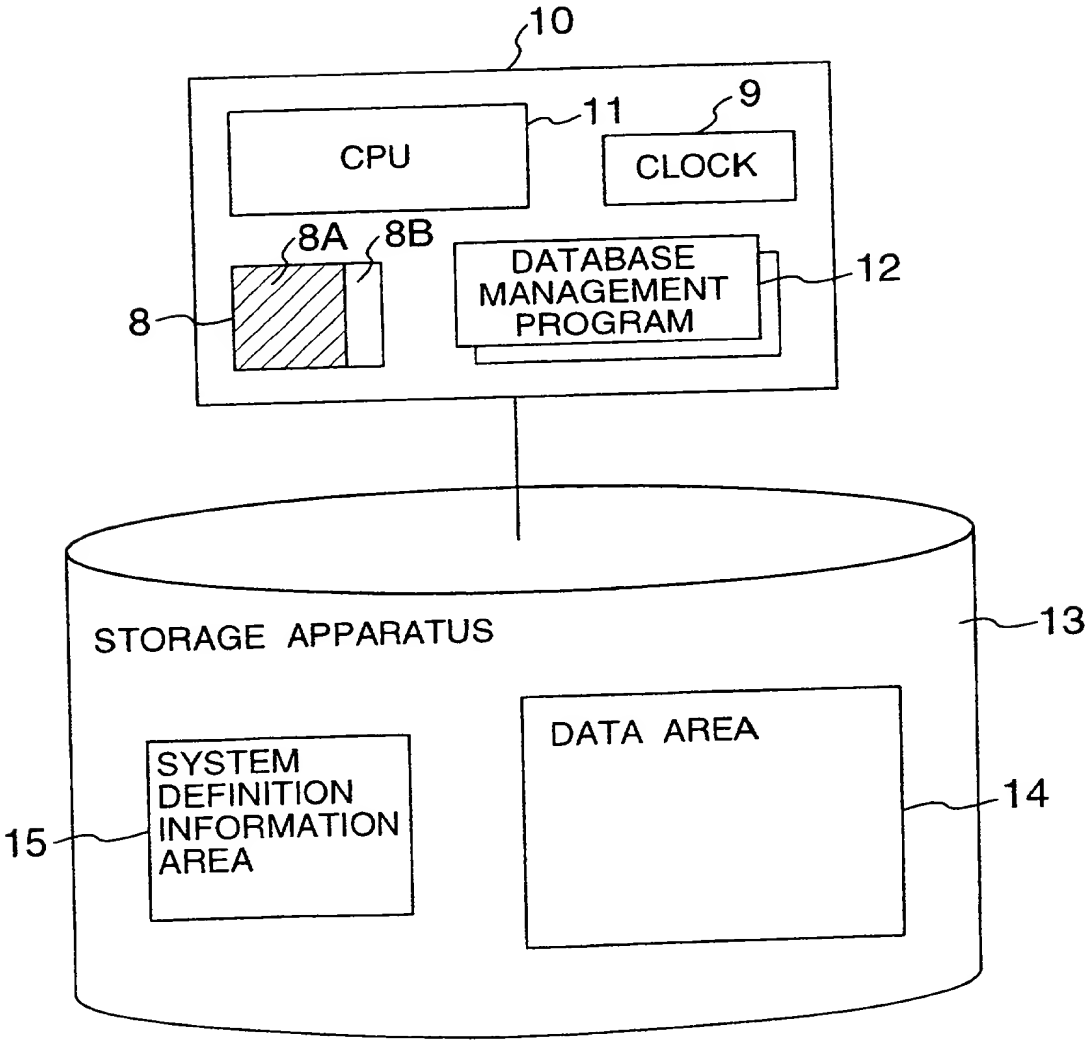

10
CPU
11
9
CLOCK
8A
8B
8
DATABASE
MANAGEMENT
PROGRAM
12
STORAGE APPARATUS
13
DATA AREA
14
SYSTEM
DEFINITION
INFORMATION
AREA
15

FIG. 3

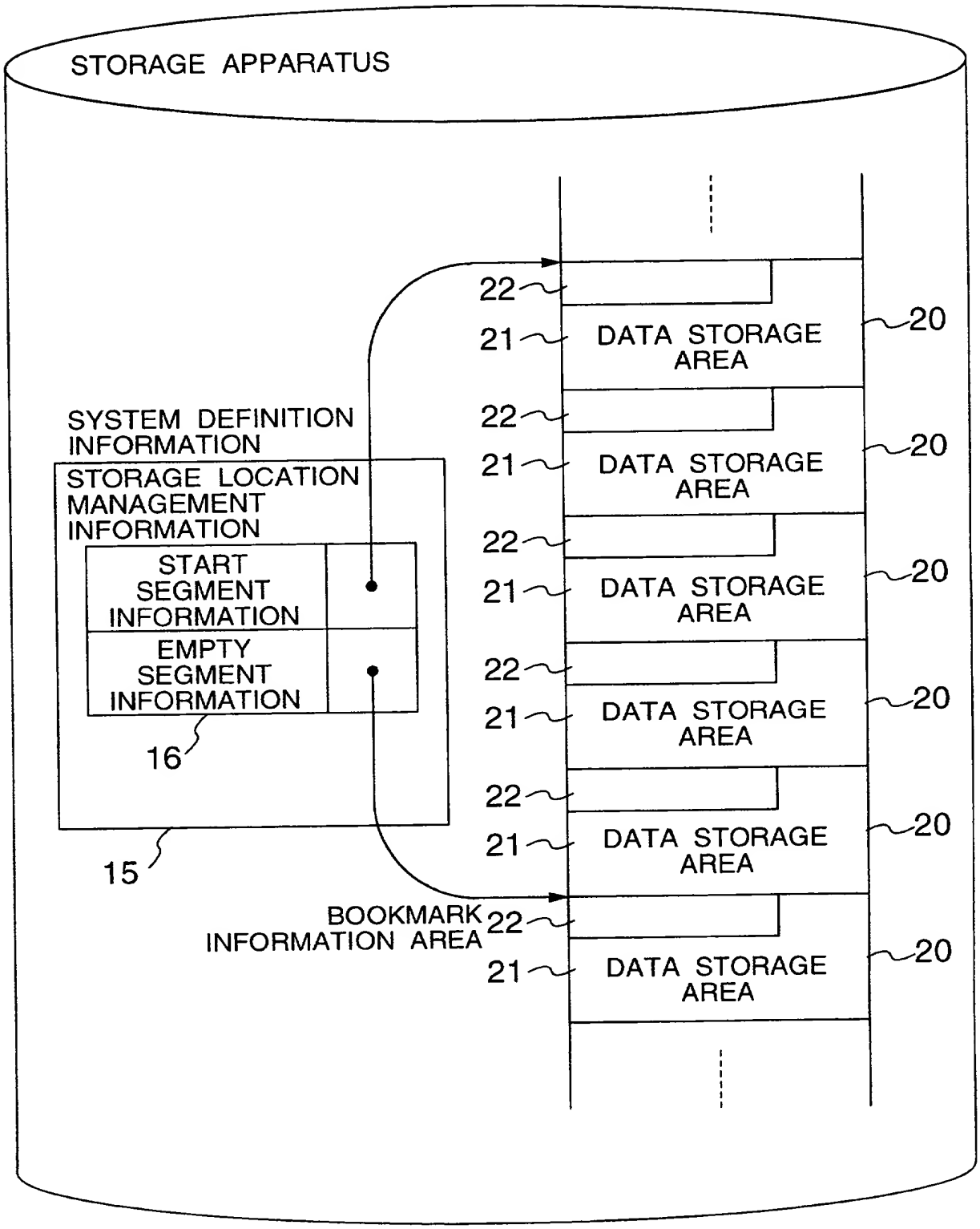

STORAGE APPARATUS
SYSTEM DEFINITION INFORMATION
STORAGE LOCATION MANAGEMENT INFORMATION
START SEGMENT INFORMATION
EMPTY SEGMENT INFORMATION
16
15
BOOKMARK INFORMATION AREA
22
21
DATA STORAGE AREA
20

FIG. 4

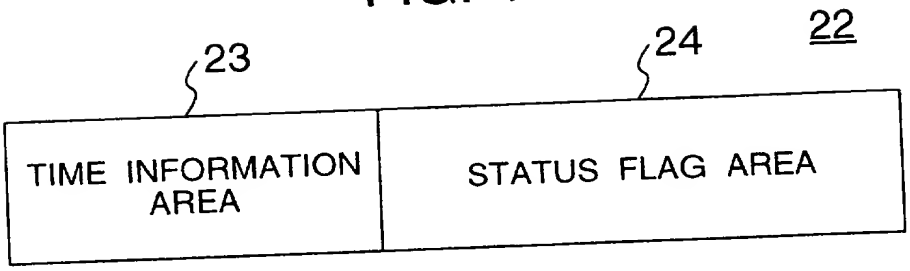

23
24
22
TIME INFORMATION AREA
STATUS FLAG AREA

FIG. 6

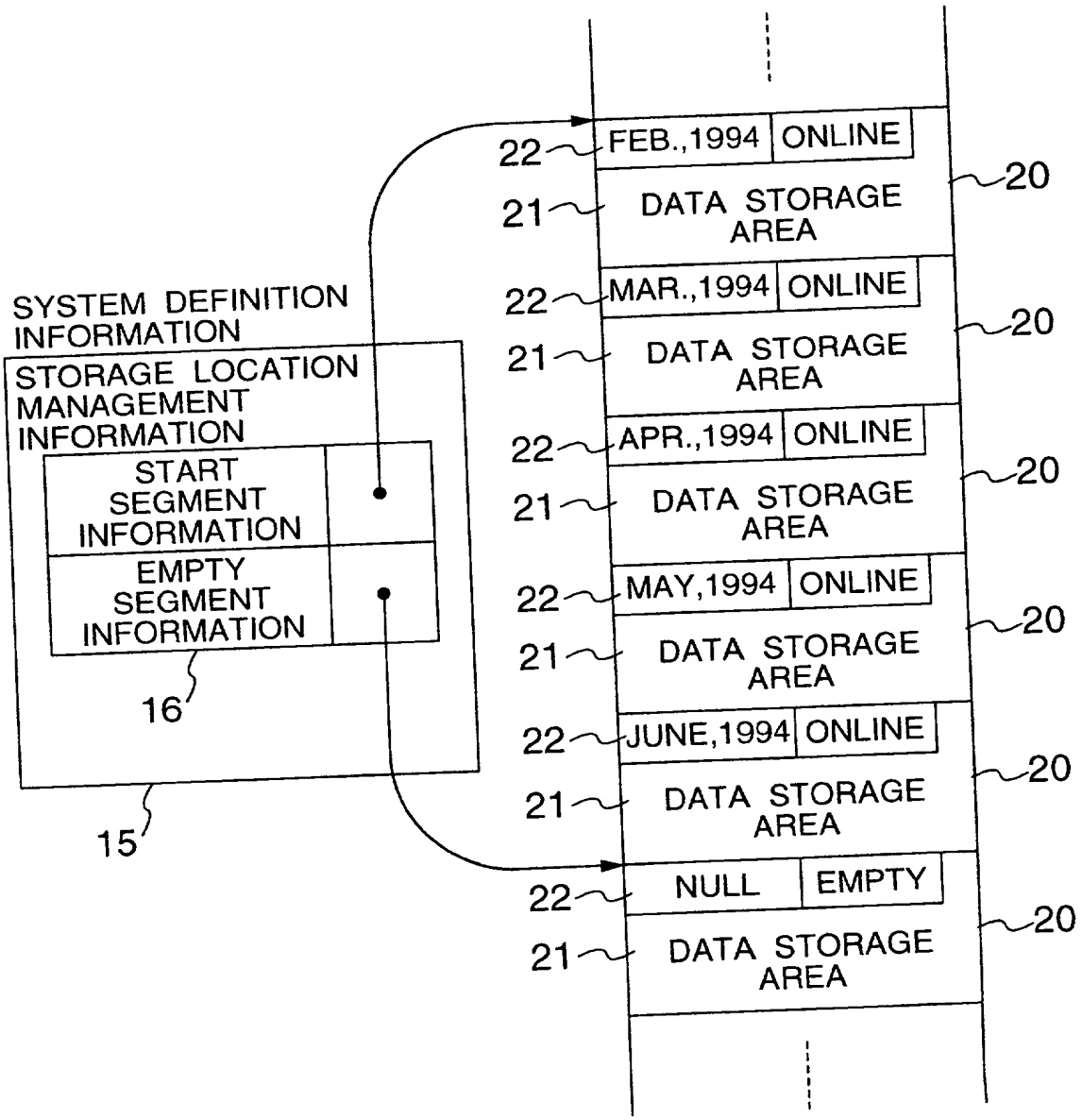

SYSTEM DEFINITION INFORMATION
STORAGE LOCATION MANAGEMENT INFORMATION
START SEGMENT INFORMATION
EMPTY SEGMENT INFORMATION
16
15
22
21
20
FEB.,1994
ONLINE
DATA STORAGE AREA
MAR.,1994
APR.,1994
MAY,1994
JUNE,1994
NULL
EMPTY

FIG. 5

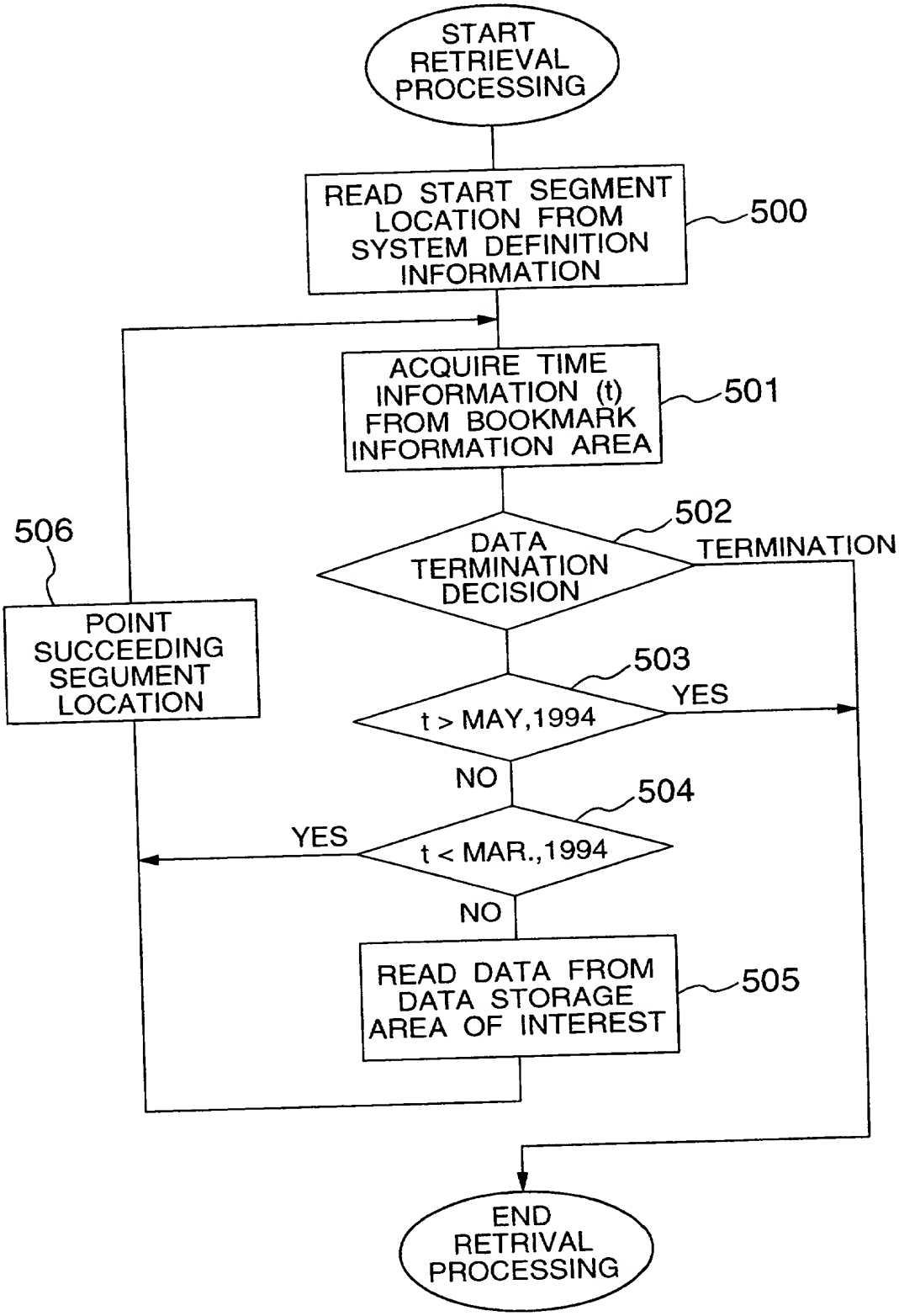

START RETRIEVAL PROCESSING
READ START SEGMENT LOCATION FROM SYSTEM DEFINITION INFORMATION
500
ACQUIRE TIME INFORMATION (t) FROM BOOKMARK INFORMATION AREA
501
DATA TERMINATION DECISION
502
TERMINATION
506
POINT SUCCEEDING SEGUMENT LOCATION
t > MAY,1994
503
YES
NO
t < MAR.,1994
504
YES
NO
READ DATA FROM DATA STORAGE AREA OF INTEREST
505
END RETRIVAL PROCESSING

FIG. 7

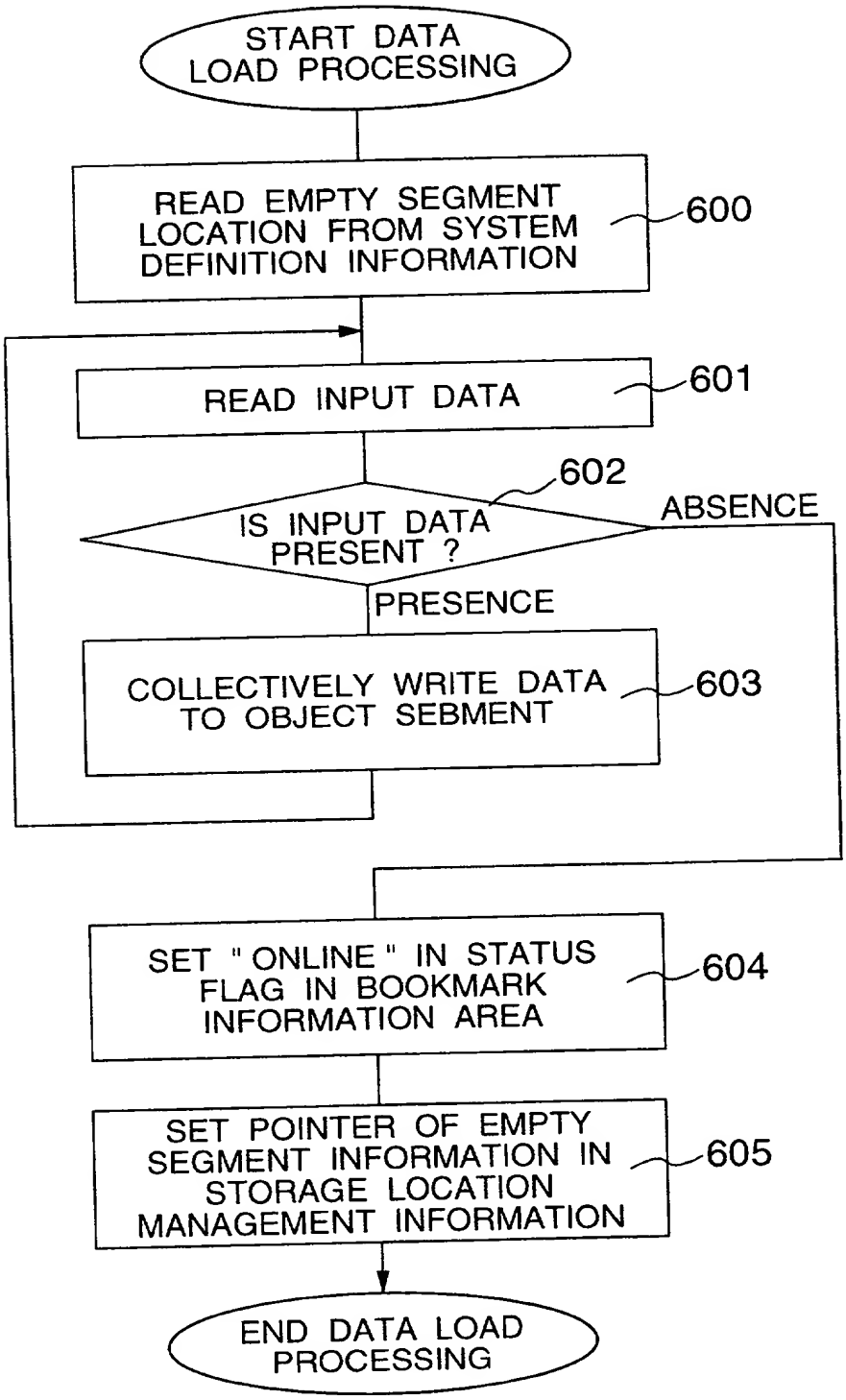

START DATA LOAD PROCESSING
READ EMPTY SEGMENT LOCATION FROM SYSTEM DEFINITION INFORMATION
600
READ INPUT DATA
601
602
IS INPUT DATA PRESENT ?
ABSENCE
PRESENCE
COLLECTIVELY WRITE DATA TO OBJECT SEBMENT
603
SET " ONLINE " IN STATUS FLAG IN BOOKMARK INFORMATION AREA
604
SET POINTER OF EMPTY SEGMENT INFORMATION IN STORAGE LOCATION MANAGEMENT INFORMATION
605
END DATA LOAD PROCESSING

FIG. 8

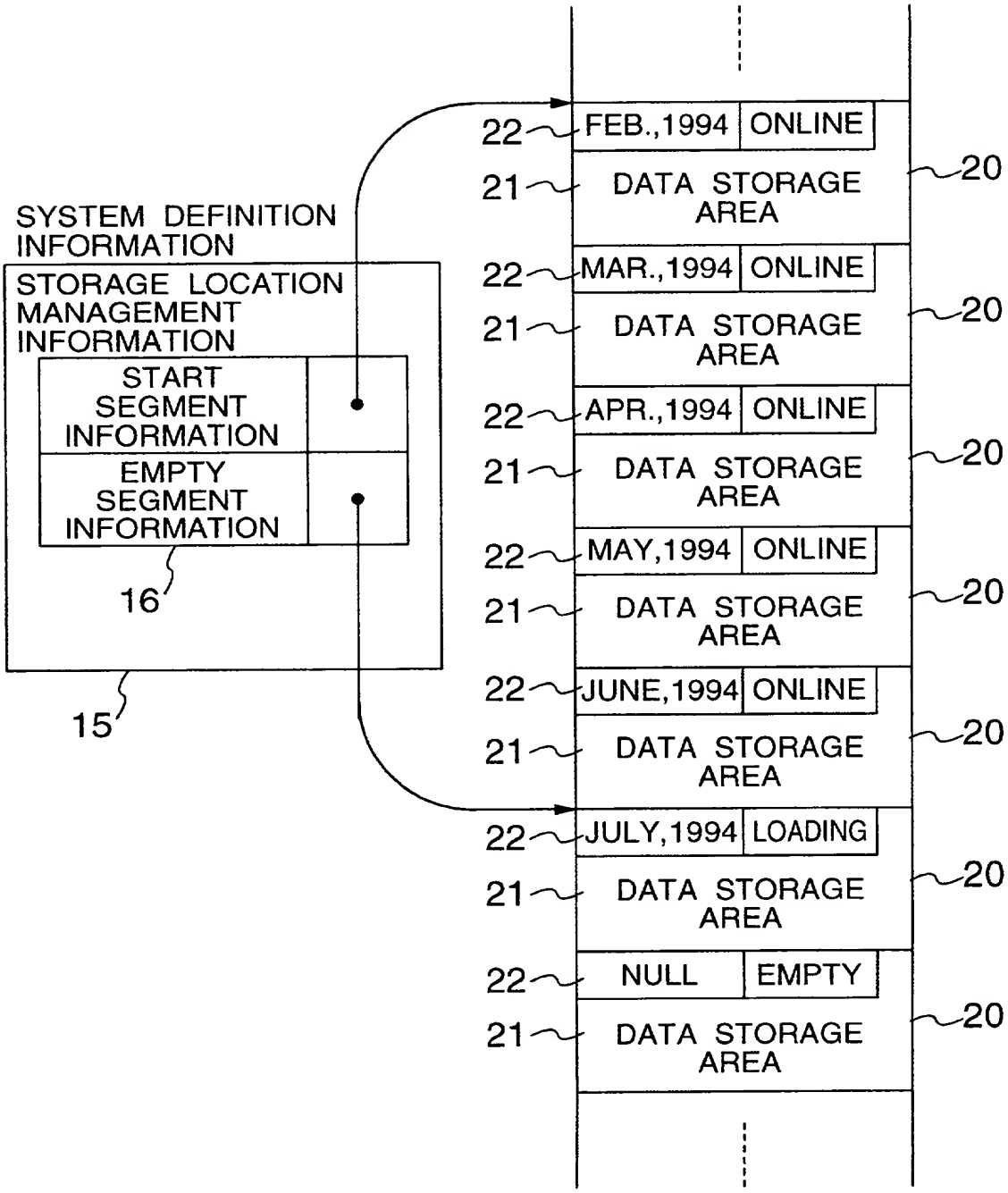

SYSTEM DEFINITION INFORMATION
STORAGE LOCATION MANAGEMENT INFORMATION
START SEGMENT INFORMATION
EMPTY SEGMENT INFORMATION
16
15
22
FEB.,1994
ONLINE
21
DATA STORAGE AREA
20
MAR.,1994
APR.,1994
MAY,1994
JUNE,1994
JULY,1994
LOADING
NULL
EMPTY

FIG. 9

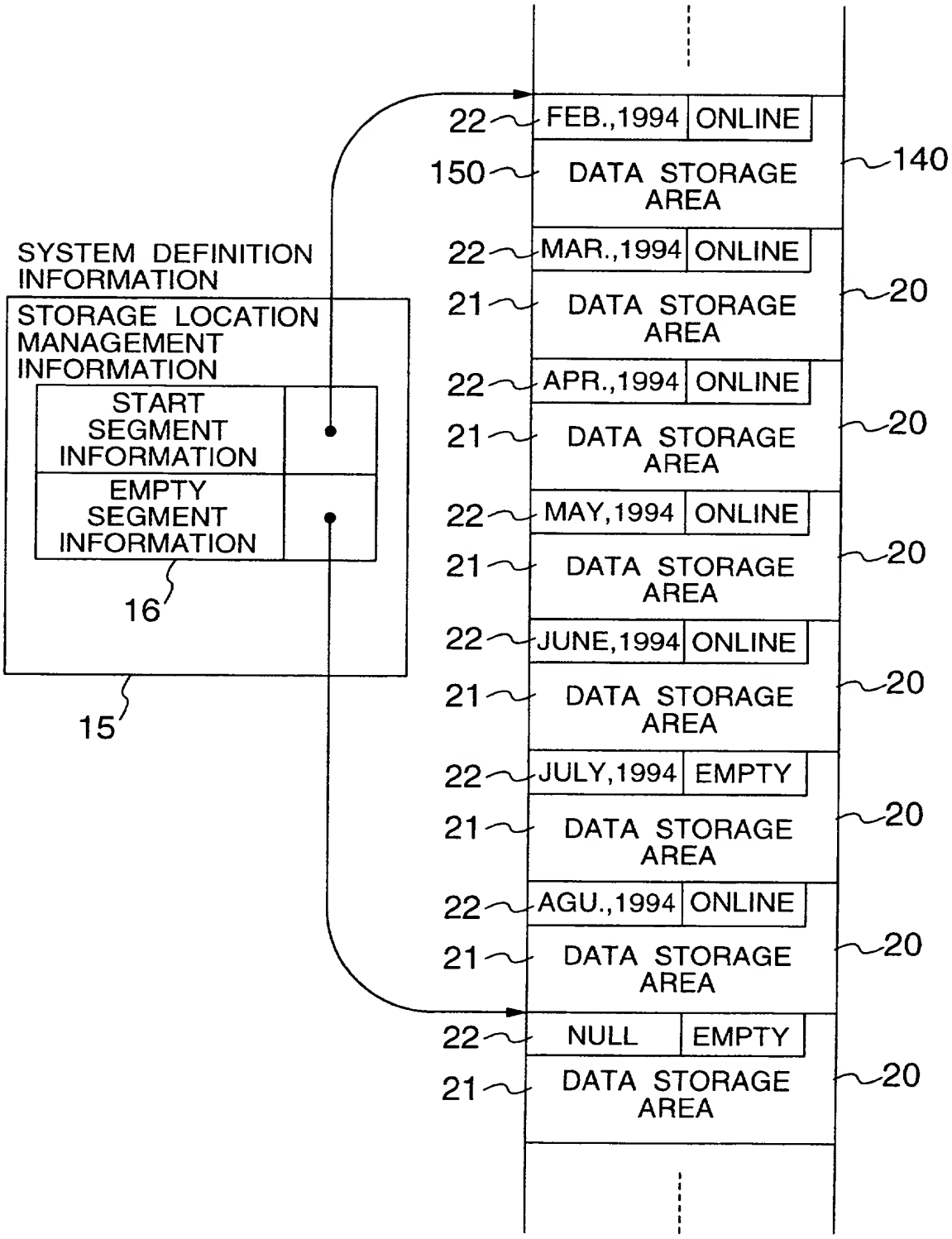

SYSTEM DEFINITION INFORMATION
STORAGE LOCATION MANAGEMENT INFORMATION
START SEGMENT INFORMATION
EMPTY SEGMENT INFORMATION
16
15
22
FEB.,1994
ONLINE
150
DATA STORAGE AREA
140
MAR.,1994
21
20
APR.,1994
MAY,1994
JUNE,1994
JULY,1994
EMPTY
AGU.,1994
NULL

FIG. 10

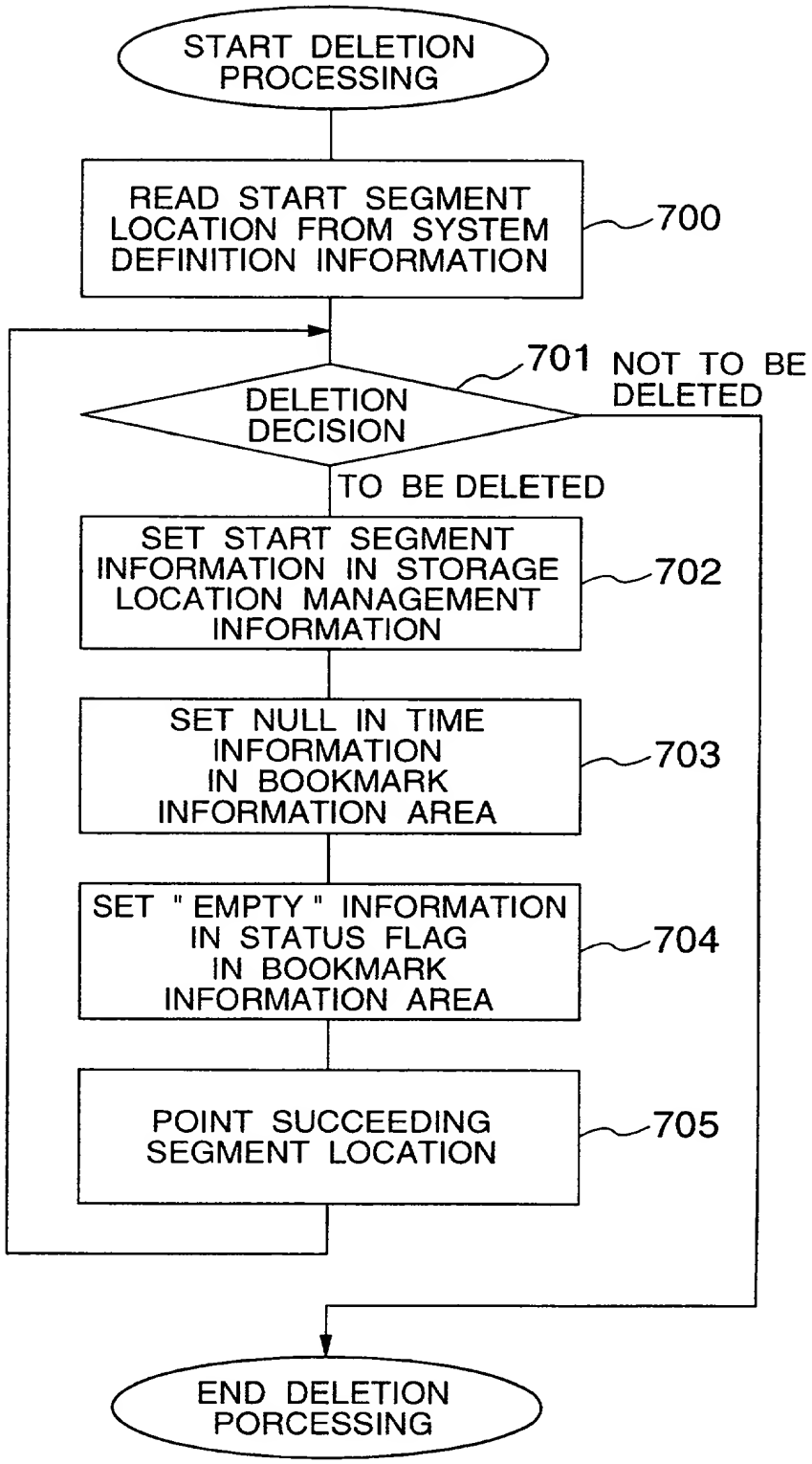

START DELETION PROCESSING
READ START SEGMENT LOCATION FROM SYSTEM DEFINITION INFORMATION
700
DELETION DECISION
701
NOT TO BE DELETED
TO BE DELETED
SET START SEGMENT INFORMATION IN STORAGE LOCATION MANAGEMENT INFORMATION
702
SET NULL IN TIME INFORMATION IN BOOKMARK INFORMATION AREA
703
SET " EMPTY " INFORMATION IN STATUS FLAG IN BOOKMARK INFORMATION AREA
704
POINT SUCCEEDING SEGMENT LOCATION
705
END DELETION PORCESSING

FIG. 11

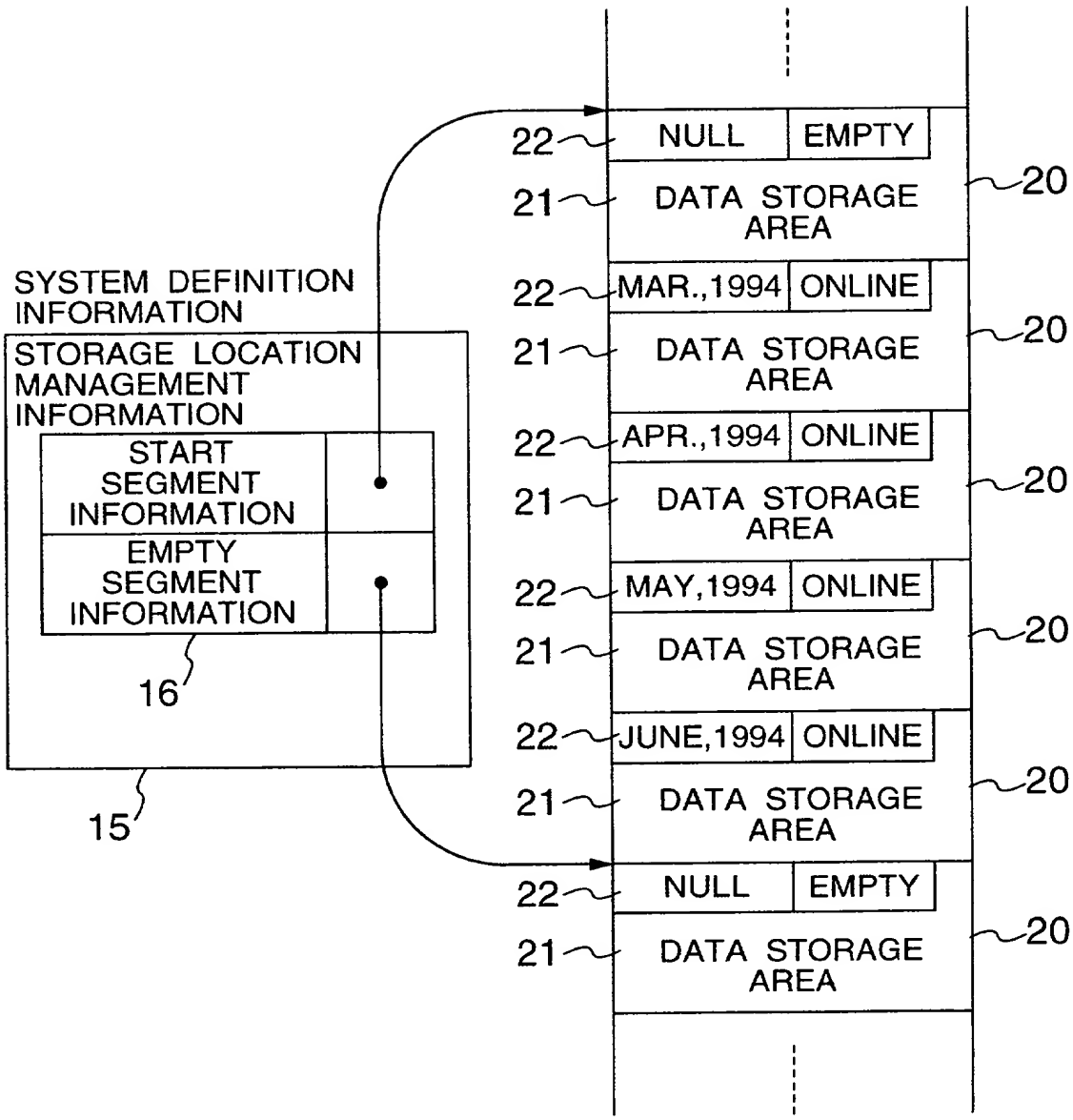

SYSTEM DEFINITION INFORMATION
STORAGE LOCATION MANAGEMENT INFORMATION
START SEGMENT INFORMATION
EMPTY SEGMENT INFORMATION
16
15
22
NULL
EMPTY
21
DATA STORAGE AREA
20
MAR.,1994
ONLINE
APR.,1994
MAY,1994
JUNE,1994

FIG. 12

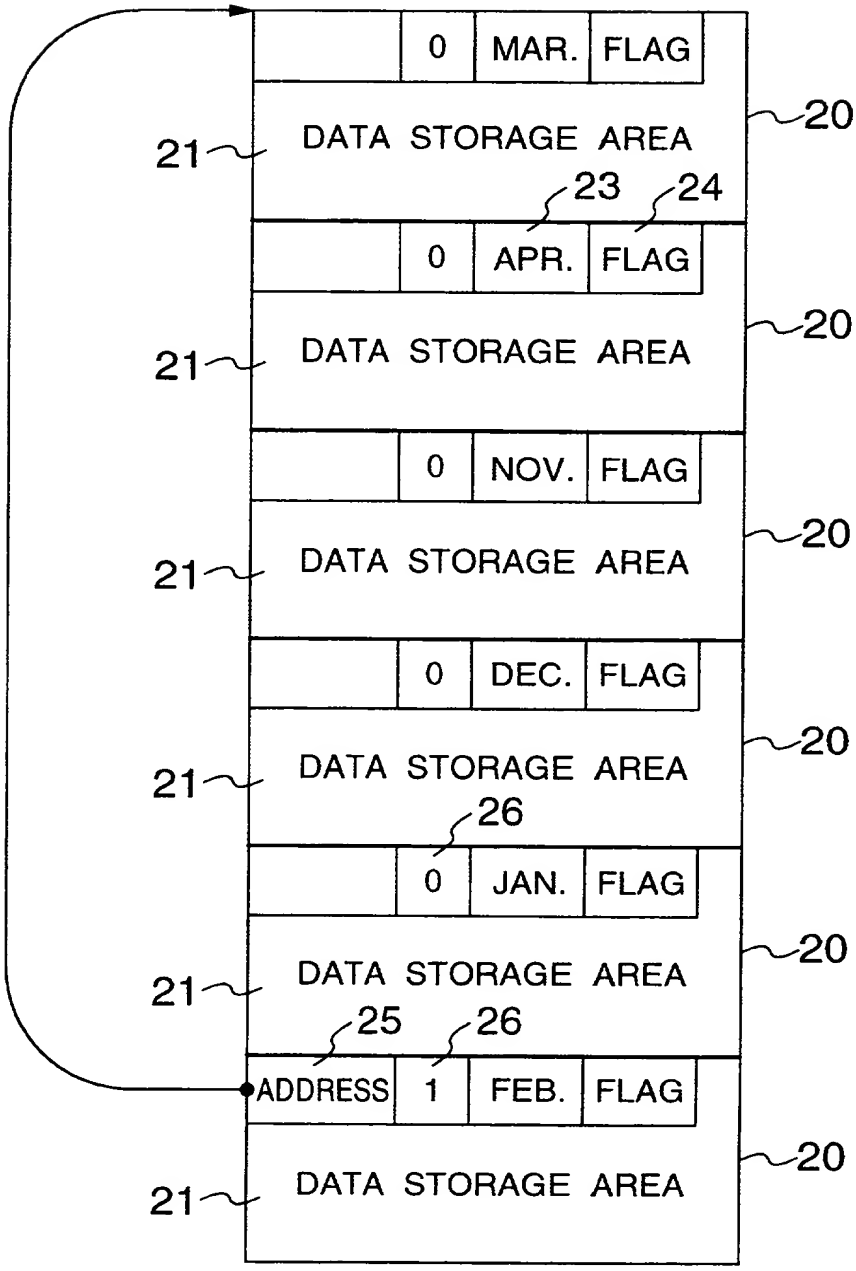
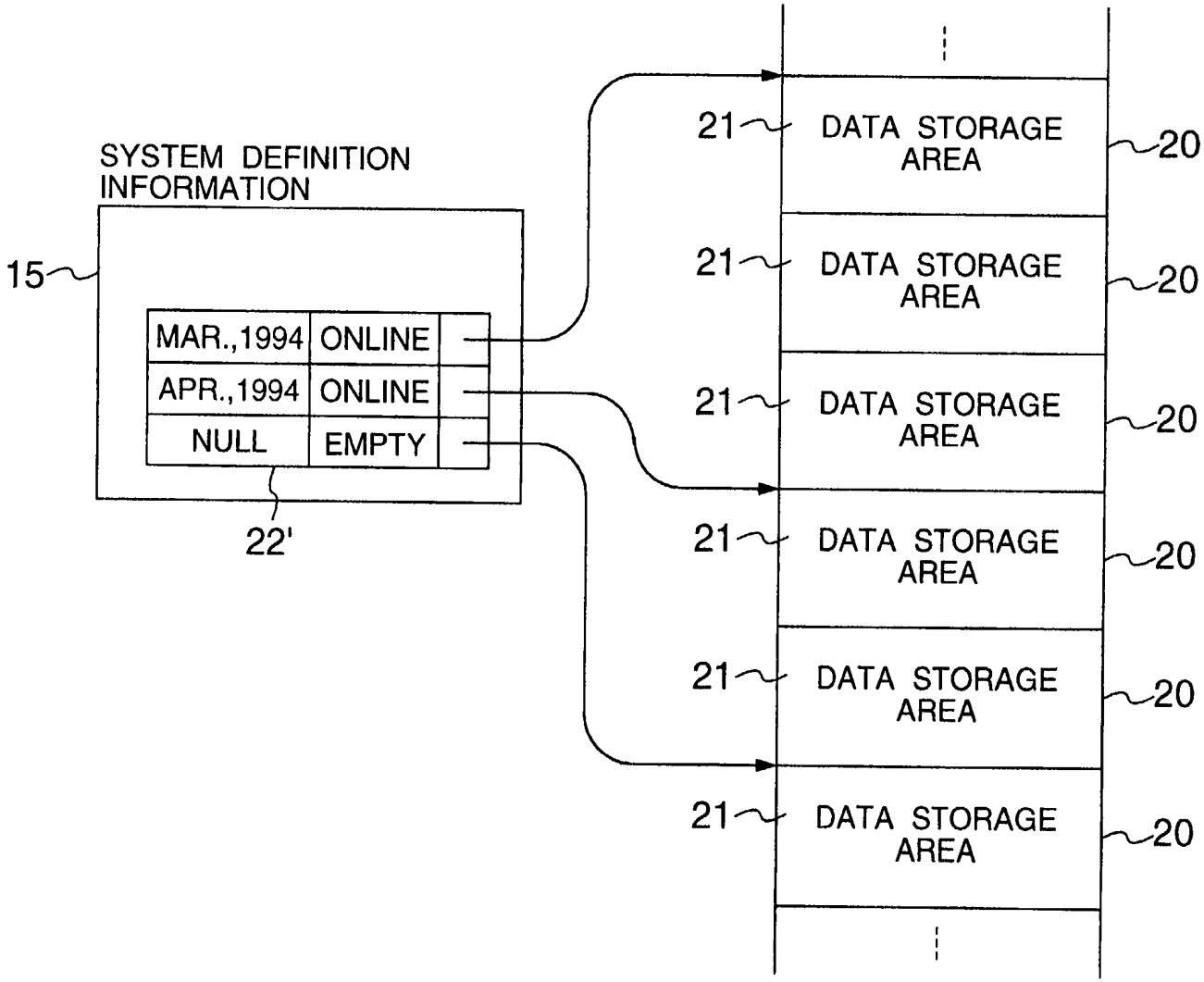

0
MAR.
FLAG
DATA STORAGE AREA
20
21
23
24
0
APR.
FLAG
DATA STORAGE AREA
20
21
0
NOV.
FLAG
DATA STORAGE AREA
20
21
0
DEC.
FLAG
DATA STORAGE AREA
20
21
26
0
JAN.
FLAG
DATA STORAGE AREA
20
21
25
26
ADDRESS
1
FEB.
FLAG
DATA STORAGE AREA
20
21

FIG. 13


SYSTEM DEFINITION INFORMATION
15
MAR.,1994
ONLINE
APR.,1994
ONLINE
NULL
EMPTY
22'
21
DATA STORAGE AREA
20
21
DATA STORAGE AREA
20
21
DATA STORAGE AREA
20
21
DATA STORAGE AREA
20
21
DATA STORAGE AREA
20
21
DATA STORAGE AREA
20

E3750-01
(*)

DECLARATION AND POWER OF ATTORNEY FOR PATENT APPLICATION

As a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below next to my name, I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled:

"DATA STRUCTURE IN DATABASE, DATABASE SYSTEM FOR MANAGING DATABASE AND DATABASE MANAGING METHOD AND SYSTEM",

the specification of which (check one)

[X] is attached hereto.

[ ] was filed on _____
as Application Serial No. _____
and was amended on _____
(if applicable)

I hereby state that I have reviewed and understand the contents of the above-identified specification, including the claims, as amended by any amendment referred to above.

I acknowledge the duty to disclose information which is material to the examination of this application in accordance with Title 37, Code of Federal Regulations, §1.56(a).

I hereby claim foreign priority benefits under Title 35, United States Code, §119 of any foreign application(s) for patent or inventor's certificate listed below and have also identified below any foreign application for patent or inventor's certificate having a filing date before that of the application on which priority is claimed:

Prior Foreign Application(s)			Priority Claimed	
09-065919 (Number)	Japan (Country)	19 March, 1997 (Day/Month/Year Filed)	[X] Yes	[ ] No
(Number)	(Country)	(Day/Month/Year Filed)	[ ] Yes	[ ] No
(Number)	(Country)	(Day/Month/Year Filed)	[ ] Yes	[ ] No
(Number)	(Country)	(Day/Month/Year Filed)	[ ] Yes	[ ] No
(Number)	(Country)	(Day/Month/Year Filed)	[ ] Yes	[ ] No
(Number)	(Country)	(Day/Month/Year Filed)	[ ] Yes	[ ] No

I hereby claim the benefit under Title 35, United States Code, §120 of any United States application(s) listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States application in the manner provided by the first paragraph of Title 35, United States Code, §112, I acknowledge the duty to disclose material information as defined in Title 37, Code of Federal Regulations, §1.56(a) which occurred between the filing date of the prior application and the national or PCT international filing date of this application:

(Application Serial No.)	(Filing Date)	(Status: patented, pending, abandoned)
(Application Serial No.)	(Filing Date)	(Status: patented, pending, abandoned)
(Application Serial No.)	(Filing Date)	(Status: patented, pending, abandoned)
(Application Serial No.)	(Filing Date)	(Status: patented, pending, abandoned)

(Continued on Page 2)

E3750-01

I hereby appoint as principal attorneys; Donald R. Antonelli, Reg. No. 20,296; David T. Terry, Reg. No. 20,178; Melvin Kraus, Reg. No. 22,466; Stanley A. Wal, Reg. No. 26,432; William I. Solomon, Reg. No. 28,565; Gregory E. Montone, Reg. No. 28,141; Ronald J. Shore, Reg. No. 28,577; Donald E. Stout, Reg. No. 26,422; Alan E. Schiavelli, Reg. No. 32,087; James N. Dresser, Reg. No. 22,973 and Carl I. Brundidge, Reg. No. 29,621 to prosecute and transact all business connected with this application and any related United States application and international applications. Please direct all communications to the following address:

Antonelli, Terry, Stout & Kraus
Suite 1800
1300 North Seventeenth Street
Arlington, Virginia 22209
Telephone: (703) 312-6600
Fax: (703) 312-6666

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further, that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United State Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

(Full Name) (Signature)

Date March 6, 1998 Inventor Takayoshi SHIMOKAWA Takayoshi Shimokawa

Residence Yokohama-shi, Japan Citizenship Japan

Post Office Address 14-1, Nakatakita-1-chome, Izumi-ku, Yokohama-shi, Japan.

Date March 6, 1998 Inventor Kazuo MASAI Kazuo Masai

Residence Yokohama-shi, Japan Citizenship Japan

Post Office Address 31-14, Hino-7-chome, Konan-ku, Yokohama-shi, Japan.

Date _____ Inventor _____

Residence _____ Citizenship _____

Post Office Address _____

Date _____ Inventor _____

Residence _____ Citizenship _____

Post Office Address _____

Date _____ Inventor _____

Residence _____ Citizenship _____

Post Office Address _____

Date _____ Inventor _____

Residence _____ Citizenship _____

Post Office Address _____

Date _____ Inventor _____

Residence _____ Citizenship _____

Post Office Address _____

Date _____ Inventor _____

Residence _____ Citizenship _____

Post Office Address _____

Date _____ Inventor _____

Residence _____ Citizenship _____

Post Office Address _____

Date _____ Inventor _____

Residence _____ Citizenship _____

Post Office Address _____